

Freigabestände von strukturierten Objektversionsmengen in Bauprojekten

Daniel G. Beer, Berthold Firmenich
{daniel.beer | berthold.firmenich}@bauing.uni-weimar.de

Bauhaus-Universität Weimar

1 Einleitung

Planungsphase: Die Planungsphase von Bauprojekten ist in Grundzügen vergleichbar mit der Software-Entwicklung. In beiden sind zahlreiche Beteiligte damit beschäftigt, Planungsmaterial bzw. Komponenten zu erzeugen, zu ändern oder auszutauschen, wobei alle an einem gemeinsamen Ziel arbeiten. Die Organisation und Kontrolle von Änderungen wird in der Softwareentwicklung mit Software-Konfigurationsmanagement bezeichnet. Bei Bauprojekten spricht man vom Projektmanagement.

Produktmodell: Es wird vorausgesetzt, dass das gemeinsame Planungsmaterial eines Bauprojekts objektorientiert in Form von Klassen, Objekten und deren Abhängigkeiten vorliegt. Das Planungsmaterial wird dann als Produkt- bzw. Bauwerksmodell bezeichnet. Es wird während der Planung von verschiedenen Planern teilweise parallel verändert. Aus rechtlichen und praktischen Gründen dürfen nicht alle Zustände des Planungsmaterials überschrieben werden, so dass das Produktmodell versioniert sein muss [Firmenich 2002].

Prozessmodell: Der Ablauf der Planungsphase kann mit Hilfe von Prozessmodellen beschrieben werden. Diese werden vom Projektmanagement definiert, überwacht und ggf. angepasst. Prozessmodelle stellen den Zusammenhang zwischen einzelnen Planungsaufgaben her und ermöglichen eine koordinierte Projektarbeit [Beer 2002]. Sie nutzen dabei Operationen des Produktmodells.

Freigabestand: Beim Zugriff des Prozessmodells auf das versionierte Produktmodell sind Freigabestände erforderlich. Ihre Definition ist Ziel dieses Beitrags.

2 Objektversionsmenge

Einleitung: Das versionierte Produktmodell wird als Menge von Objektversionen und den Beziehungen zwischen diesen auf Basis der Mengenlehre formal beschrieben. In diesem Beitrag wird die bestehende Beschreibung aus [Firmenich 2002] übernommen und sinnvoll erweitert.

2.1 Definitionen

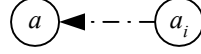
Objektmenge Ω : Die Menge aller im Projekt \mathfrak{P} enthaltenen Objekte wird als Objektmenge Ω bezeichnet.

$$\Omega := \{a \mid a \text{ ist ein Objekt im Projekt } \mathfrak{P}\} \quad (1)$$

Objektversionsmenge M : Die Menge aller Versionen, die von Objekten in Ω abgeleitet sind, wird als Objektversionsmenge M bezeichnet. Die Abbildung $\omega : M \rightarrow \Omega$ bildet die Menge der Objektversionen auf die Menge der Objekte im Projekt ab.

$$M := \{a_i \mid \exists_{a \in \Omega} \omega(a_i) = a\} \quad (2)$$

Die Abbildung ω wird durch eine strichpunktierte Kante mit gefüllter Pfeilspitze dargestellt.



Gelöschte Objektversion: Eine Objektversion darf nicht gelöscht werden [Firmenich 2002]. Deshalb werden virtuelle Objektversionen $\delta \in \Delta$ eingeführt, die Objektversionen als gelöscht markieren.

$$\Delta := \{\delta \mid \delta \text{ kennzeichnet eine Objektversion als gelöscht}\} \quad (3)$$

$$M_\Delta := M \cup \Delta \quad (4)$$

Eine virtuelle Objektversion wird durch ein spezielles Symbol dargestellt.



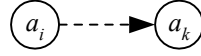
Objektversionsrelation V : Die Beziehungen zwischen Objektversionen aus M werden mit der Objektversionsrelation $V \subseteq (M_\Delta \times M_\Delta) \setminus (\Delta \times \Delta)$ beschrieben. Sie enthält die Paare von Objektversionen, bei denen der Nachfolger aus dem Vorgänger durch eine Änderung hervorgegangen ist.

$$V := \{(a_i, a_j) \in (M_\Delta \times M_\Delta) \setminus (\Delta \times \Delta) \mid \text{Objektversion } a_j \text{ ist Nachfolger von Objektversion } a_i\} \quad (5)$$

Objektversionsgraph \mathfrak{G}_V : Der Objektversionsgraph \mathfrak{G}_V ist die grafische Repräsentation der Objektversionsmenge M (Knotenmenge $K(\mathfrak{G}_V)$) und der Objektversionsrelation V (Kantenmenge $R(\mathfrak{G}_V)$). Er ist zyklensfrei, da vorhandene Objektversionen nicht noch einmal erzeugt werden können. Damit gibt es keinen Weg von einem Knoten ausgehend wieder zu diesem zurück und folglich keine Zyklen.

$$\mathfrak{G}_V := (M; V) \quad K(\mathfrak{G}_V) := M \quad R(\mathfrak{G}_V) := V \quad (6)$$

Eine Kante wird im Objektversionsgraphen \mathfrak{G}_V gestrichelt mit ausgefüllter Pfeilspitze dargestellt.



Bindungsrelation B : Ist eine Objektversion $b_k \in M$ an eine Objektversion $a_i \in M$ gebunden, so wird das geordnete Paar $(a_i, b_k) \in M \times M$ in die Menge B eingetragen [Pahl und Beucke 2000].

$$B := \{(a_i, b_k) \in M \times M \mid \text{Objektversion } a_i \text{ bindet Objektversion } b_k\} \quad (7)$$

Bindungsgraph \mathfrak{G}_B : Der Bindungsgraph \mathfrak{G}_B ist die grafische Repräsentation der Objektversionsmenge M (Knotenmenge $K(\mathfrak{G}_B)$) und der Bindungsrelation B (Kantenmenge $R(\mathfrak{G}_B)$).

$$\mathfrak{G}_B := (M; B) \quad K(\mathfrak{G}_B) := M \quad R(\mathfrak{G}_B) := B \quad (8)$$

Eine Kante gebundener Objektversionen wird als Volllinie mit ausgefüllter Pfeilspitze dargestellt.

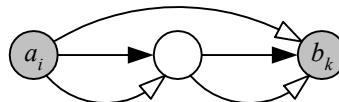


2.2 Begriffe

Abhängige Objektversionen: Existiert ein Weg $\langle a_i, \dots, b_k \rangle$ im Bindungsgraphen \mathfrak{G}_B , so heißt b_k von a_i abhängig. Es existiert eine Kante $(a_i, b_k) \in \langle B \rangle_t$ in der transitiven Hülle der Bindungen B .

$$b_k \text{ ist abhängig von } a_i \Leftrightarrow (a_i, b_k) \in \langle B \rangle_t \quad (9)$$

Eine Kante abhängiger Objektversionen wird als Volllinie mit leerer Pfeilspitze dargestellt.



Ursprungsversion: Eine Ursprungsversion $a_i \in M$ ist die erste Objektversion eines Objekts $a \in \Omega$. Für ein Objekt a gibt es genau eine Ursprungsversion a_i .

$$a_i \in M \text{ ist die Ursprungsversion des Objekts } a \in \Omega :\Leftrightarrow \omega(a_i) = a \wedge \exists_{\delta \in \Delta} ((\delta, a_i) \in V) \quad (10)$$

Variante: Eine Nachfolgeversion $a_j \in M$ der Objektversion $a_i \in M$ wird Variante von a_i genannt, wenn a_i mehrere direkte Nachfolger hat.

$$a_j \in M \text{ ist eine Variante von } a_i \in M :\Leftrightarrow (a_i, a_j) \in V \wedge \exists_{a_k \in M} (a_j \neq a_k \wedge (a_i, a_k) \in V) \quad (11)$$

Revision: Eine Nachfolgeversion $a_j \in M$ der Objektversion $a_i \in M$ wird Revision von a_i genannt, wenn a_i nur einen direkten Nachfolger a_j hat.

$$a_j \in M \text{ ist eine Revision von } a_i \in M :\Leftrightarrow (a_i, a_j) \in V \wedge \forall_{a_k \in M} (a_j \neq a_k \wedge (a_i, a_k) \notin V) \quad (12)$$

Zusammenführung: Eine Objektversion $a_k \in M$, die aus allen Objektversionen $a_i \in M_i \subseteq M$ hervorgeht, heißt Zusammenführung der Objektversionen in M_i .

$$a_k \in M \text{ ist die Zusammenführung der Objektversionen } a_i \in M_i :\Leftrightarrow \quad (13)$$

$$\forall_{a_i \in M_i} \forall_{a_j \in M_i} ((a_i, a_k) \in V \wedge (a_j, a_k) \in V \wedge a_i \neq a_j)$$

Überholte Objektversion: Eine Objektversion $a_i \in M$ ist überholt, wenn es mindestens eine neuere Objektversion $a_j \in M_\Delta$ des gleichen Objekts $a \in \Omega$ gibt (Variante oder Revision).

$$a_i \text{ ist überholt} :\Leftrightarrow \exists_{a_j \in M_\Delta} ((a_i, a_j) \in V) \quad (14)$$

Obsolete Objektversion: Ist eine Objektversion $b_k \in M$ an eine überholte Objektversion $a_i \in M$ gebunden, so ist sie obsolet bezüglich a_i .

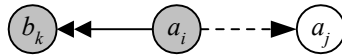
$$b_k \text{ ist obsolet} :\Leftrightarrow \exists_{a_i \in M} \exists_{a_j \in M_\Delta} ((a_i, b_k) \in B \wedge (a_i, a_j) \in V) \quad (15)$$



Menge der eingefrorenen Bindungen: Bindungen $(a_i, b_k) \in B$, die trotz obsoletter Objektversion $b_k \in M$ bezüglich $a_i \in M$ als aktuell betrachtet werden sollen, müssen „eingefroren“ werden.

$$G := \{(a_i, b_k) \in B \mid (a_i, b_k) \text{ ist eingefroren}\} \subseteq B \quad (16)$$

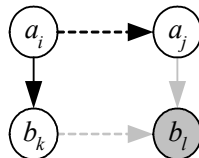
Grafisch wird eine eingefrorene Bindung als eine Vollinie mit gefülltem Doppelpfeil dargestellt.



2.3 Operationen

Objektversion modernisieren: Eine bezüglich $a_i \in M$ obsolete Objektversion $b_k \in M$ kann durch das Erzeugen einer neuen Objektversion $b_l \in M$ und Binden an die Nachfolgeversion $a_j \in M$ von a_i aktualisiert werden. Dieser Vorgang wird als Modernisieren bezeichnet.

$$M := M \cup \{b_l\} \quad V := V \sqcup \{(b_k, b_l)\} \quad B := B \sqcup \{(a_j, b_l)\} \quad (17)$$



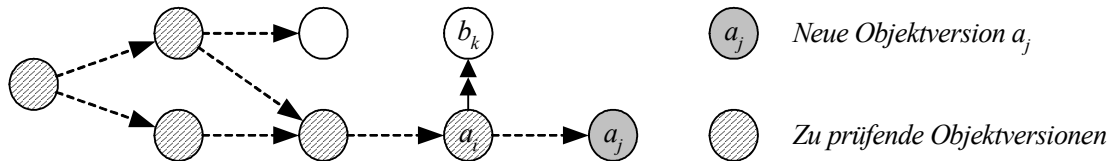
Bindung einfrieren: Soll eine Bindung (a_i, b_k) „eingefroren“ werden, so ist sie in der Relation $G \sqsubseteq B$ zu speichern.

$$G := G \sqcup \{(a_i, b_k)\} \quad (18)$$

Neue Objektversion: Wird eine neue Objektversion $a_j \in M$ des Objekts $a \in \Omega$ von der Objektversion $a_i \in M$ erzeugt, so muss diese in die Objektversionsmenge, das Tupel (a_i, a_j) in die Objektversionsrelation und das Tupel (a_j, a) in die Abbildung ω eingetragen werden.

$$M := M \cup \{a_j\} \qquad V := V \sqcup \{(a_i, a_j)\} \qquad w := w \sqcup \{(a_j, a)\} \qquad (19)$$

Durch das Eintragen einer neuen Objektversion a_j wird die Aktualität aller eingefrorenen Bindungen G bzw. Abhängigkeiten $\langle G \rangle_t$, die von Vorgängerversionen von a_j ausgehen, in Frage gestellt. Bei Verzweigungen des Objektversionsgraphen \mathfrak{G}_V sind jeweils nur die Vorgänger und keine Nachfolger zu prüfen. Es sind alle Wege von der neu erzeugten Objektversion bis zur Ursprungsversion zu betrachten.



Die Entscheidung, ob eingefrorene Bindungen $(a_i, b_k) \in G$ weiterhin aktuell sind, muss durch den Planer, der die Datenhoheit über $b_k \in M$ hat, getroffen werden. Sind eingefrorene Bindungen nicht mehr aktuell, so müssen sie aus G ausgetragen werden. Zwei Möglichkeiten der Umsetzung sind denkbar:

- Manuelles Austragen: Beim Erzeugen von a_j muss der Planer, der die Datenhoheit über b_k hat, über die Aktualität eingefrorener Bindungen entscheiden. Das Austragen muss bestätigt werden.
- Manuelles Eintragen: Beim Erzeugen von a_j wird die eingefrorene Bindung (a_i, b_k) automatisch aus G ausgetragen. Entsprechende Bindungen müssen später manuell wieder eingefroren werden.

Objektversion löschen: Durch das Löschen einer Objektversion $a_i \in M$ wird ggf. die Aktualität von eingefrorenen Bindungen in Frage gestellt (siehe *Neue Objektversion*).

$$\Delta := \Delta \cup \{\delta_i\} \qquad V := V \sqcup \{(a_i, \delta_i)\} \qquad (20)$$

3 Freigabestand

Einleitung: Für die Objektversionsmenge mit Abhängigkeiten werden im folgenden Eigenschaften, die zur Definition gültiger Freigabestände erfüllt sein müssen, definiert.

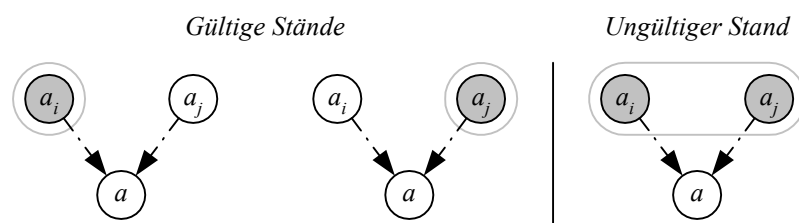
3.1 Definition

Freigabestand \mathfrak{F} : \mathfrak{F} ist ein Untergraph [Pahl und Damrath 2000] von \mathfrak{G}_B mit der Knotenmenge $K(\mathfrak{F}) \subseteq K(\mathfrak{G}_B)$, der Kantenmenge $R(\mathfrak{F}) = R(\mathfrak{G}_B) \cap K(\mathfrak{F}) \times K(\mathfrak{F})$ und folgenden Eigenschaften:

Eigenschaft 1: Von jedem Objekt $a \in \Omega$ ist höchstens eine Objektversion $a_i \in M$ mit $\omega(a_i) = a$ in der Knotenmenge $K(\mathfrak{F})$ des Freigabestandes \mathfrak{F} enthalten.

$$\text{In } K(\mathfrak{F}) \text{ ist höchstens eine Objektversion eines Objekts enthalten} :\Leftrightarrow \quad (21)$$

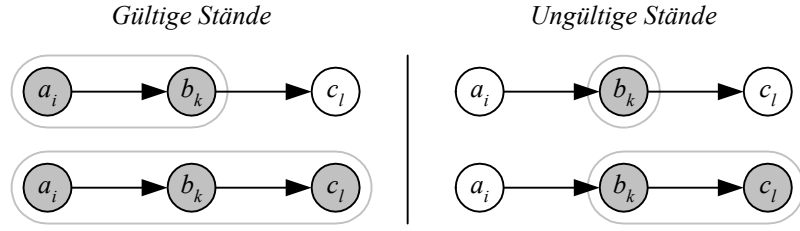
$$\bigvee_{a_i \in K(\mathfrak{F})} \bigvee_{a_k \in K(\mathfrak{F})} (\omega(a_i) = \omega(a_k) \Rightarrow a_i = a_k)$$



Eigenschaft 2: Alle Objektversionen $a_i \in M$, die Objektversionen $b_k \in K(\mathfrak{F})$ binden, müssen zur Knotenmenge des Freigabestands $K(\mathfrak{F})$ gehören.

$$K(\mathfrak{F}) \text{ enthält alle bindenden Objektversionen} :\Leftrightarrow \quad (22)$$

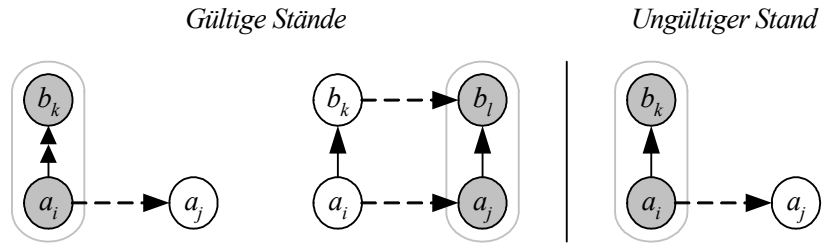
$$\forall_{a_i \in M} \quad \forall_{b_k \in K(\mathfrak{F})} \quad ((a_i, b_k) \in \langle B \rangle_t \Rightarrow a_i \in K(\mathfrak{F}))$$



Eigenschaft 3: Es sind keine obsoleten Objektversionen enthalten beziehungsweise die obsoleten Objektversionen gehören zu entsprechenden eingefrorenen Bindungen.

$$K(\mathfrak{F}) \text{ enthält keine obsoleten Objektversionen} :\Leftrightarrow \quad (23)$$

$$\forall_{b_k \in K(\mathfrak{F})} \quad \forall_{a_i \in M} \quad \forall_{a_j \in M_\Delta} \quad ((a_i, b_k) \in B \wedge (a_i, a_j) \in V \Rightarrow (a_i, b_k) \in G)$$



Freigeben: Eine Teilmenge $A \subseteq M$ der Objektversionsmenge kann selektiert und mit den inzidenten Kanten $B \sqcap A \times A$ des Bindungsgraphen \mathfrak{G}_B als Stand $\mathfrak{F} \subseteq \mathfrak{G}_B$ freigegeben werden. Die Menge muss den genannten Eigenschaften eines Freigabestands genügen. Der Stand \mathfrak{F} wird wie folgt ermittelt:

1. Selektion einer Menge A von Objektversionen, die zum Freigabestand gehören sollen. Es ist Eigenschaft 1 des Freigabestands zu erfüllen:

$$\{a \in \Omega \mid \exists_{a_i \in A} \exists_{a_k \in A} (\omega(a_i) = \omega(a_k) = a \wedge a_i \neq a_k)\} = \emptyset \quad (24)$$

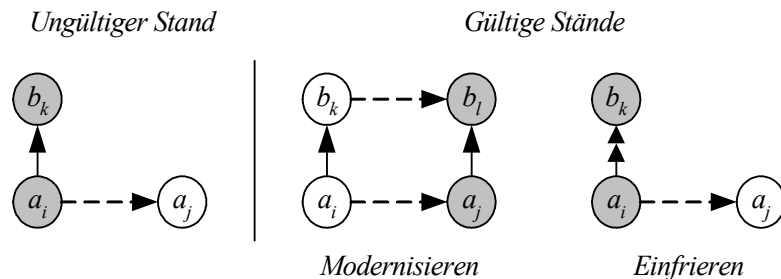
2. Eintragen aller – die selektierten Objektversionen $b_k \in A$ bindenden – Objektversionen $a_i \in M$ in die Menge A (Eigenschaft 2 des Freigabestands):

$$A := A \cup \{a_i \in M \mid \forall_{b_k \in A} (a_i, b_k) \in \langle B \rangle_t\} \quad (25)$$

3. Ermitteln der Menge A_o obsoletter Objektversionen, die nicht zu eingefrorenen Bindungen gehören.

$$A_o := \{b_k \in A \mid \exists_{a_i \in A} \exists_{a_j \in M_\Delta} ((a_i, b_k) \in B \wedge (a_i, a_j) \in V \wedge (a_i, b_k) \notin G)\} \subseteq A \quad (26)$$

Die Objekte dieser Menge müssen vom verantwortlichen Planer entweder modernisiert oder entsprechende Bindungen eingefroren werden (Eigenschaft 3 des Freigabestands):



Im Falle des Modernisierens ist das Freigeben abzuberechnen, da erst neue Objektversionen zu erzeugen sind. Werden alle obsoleten Objektversionen eingefroren, kann die Ermittlung des Freigabestands erfolgen.

4. Speichern des Freigabestands \mathfrak{F} : Die Knotenmenge $K(\mathfrak{F})$ entspricht der Menge A . Die Kantenmenge $R(\mathfrak{F})$ lässt sich aus der Knotenmenge $K(\mathfrak{F})$ und den inzidenten Kanten $B \sqcap K(\mathfrak{F}) \times K(\mathfrak{F})$ mit Hilfe der Bindungsrelation B ermitteln.

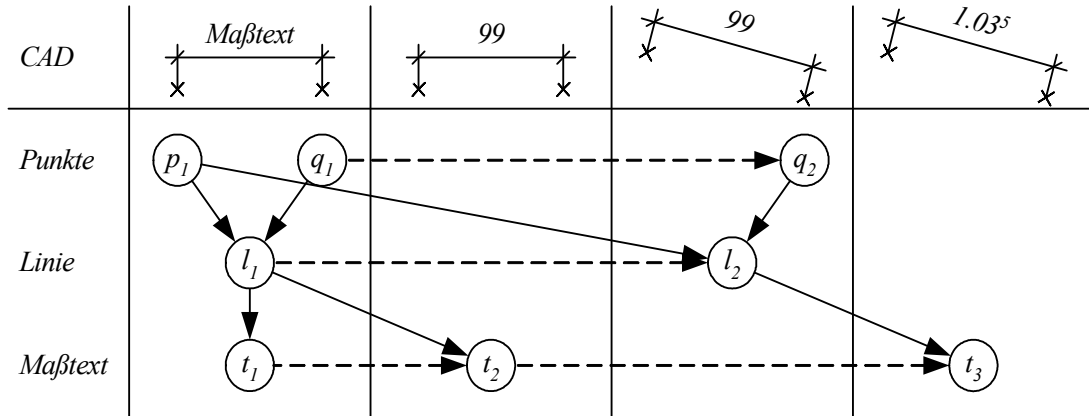
$$\mathfrak{F} := (A; B \sqcap A \times A) \quad (27)$$

3.2 Beispiel

Baubemaßung: In einer CAD-Zeichnung existiert eine assoziative Baubemaßung, die durch einen Anfangspunkt p , einen Endpunkt q , eine Maßlinie l und einen Maßtext t beschrieben wird:

$$\Omega := \{p, q, l, t\} \quad M := \{p_1, q_1, l_1, t_1\} \quad B := \{(p_1, l_1), (q_1, l_1), (l_1, t_1)\}$$

Szenario:



In dieser Zeichnung werden die folgenden Aktionen ausgeführt:

1. Ändern von t_1 (Bemaßungstext – Länge von l_1): Es entsteht die zweite Version t_2 , die an l_1 gebunden wird.

$$M := M \cup \{t_2\} \quad V := V \sqcup \{(t_1, t_2)\} \quad B := B \sqcup \{(l_1, t_2)\}$$

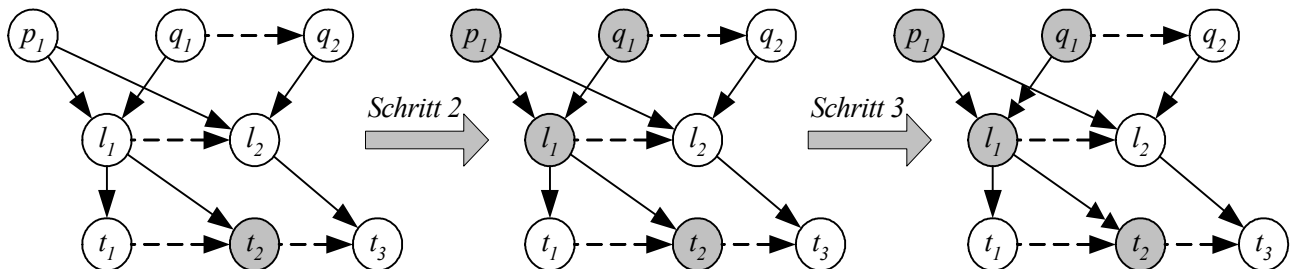
2. Verschieben des Punkts q_1 : Es entsteht die zweite Version q_2 . Die Linie l_2 wird an die Punkte p_1 und q_2 gebunden.

$$M := M \cup \{q_2, l_2\} \quad V := V \sqcup \{(q_1, q_2), (l_1, l_2)\} \quad B := B \sqcup \{(p_1, l_2), (q_2, l_2)\}$$

3. Anpassen von t_2 (Bemaßungstext – Länge von l_2): Es entsteht die Objektversion t_3 .

$$M := M \cup \{t_3\} \quad V := V \sqcup \{(t_2, t_3)\} \quad B := B \sqcup \{(l_2, t_3)\}$$

Freigabestand \mathfrak{F} : Es soll ein Freigabestand \mathfrak{F} aus der selektierten Objektversion t_2 gebildet werden.



Dazu werden die beschriebenen vier Schritte ausgeführt.

1. Selektion der Version t_2 .

$$A := \{t_2\}$$

Die Bedingung, dass nur eine Objektversion pro Objekt gewählt werden darf, ist erfüllt.

2. Die Aufnahme aller bindenden Objektversionen führt dazu, dass die Objektversionen p_1 , q_1 und l_1 mit in die Menge A aufgenommen werden müssen.

$$A := A \cup \{p_1, q_1, l_1\} = \{p_1, q_1, l_1, t_2\}$$

3. Als obsolet werden die Objektversionen t_2 bzgl. l_1 und l_1 bzgl. q_1 ermittelt. Beide obsoleten Objektversionen sollen verwendet werden. Die Bindungen (l_1, t_2) und (q_1, l_1) werden eingefroren.

$$G := G \sqcup \{(l_1, t_2), (q_1, l_1)\} = \{(l_1, t_2), (q_1, l_1)\}$$

4. Die Kantenmenge $R(\mathfrak{F})$ und der Freigabestand \mathfrak{F} ergeben sich zu:

$$\begin{aligned} R(\mathfrak{F}) &= B \sqcap A \times A \\ &= \{(p_1, l_1), (p_1, l_2), (q_1, l_1), (q_1, l_2), (l_1, t_1), (l_1, t_2), (l_2, t_3)\} \sqcap \{p_1, q_1, l_1, t_2\} \times \{p_1, q_1, l_1, t_2\} \\ &= \{(p_1, l_1), (q_1, l_1), (l_1, t_2)\} \\ \mathfrak{F} &= (\{p_1, q_1, l_1, t_2\}; \{(p_1, l_1), (q_1, l_1), (l_1, t_2)\}). \end{aligned}$$

Literatur

- [Beer 2002] BEER, D. G.: Process models as a base for communication in revitalization projects. In: *ECPPM 2002: eWork and eBusiness in Architecture, Engineering and Construction*, 2002, S. 231–236
- [Firmenich 2002] FIRMENICH, B.: *CAD im Bauplanungsprozess: Verteilte Bearbeitung einer strukturierten Menge von Objektversionen*. Aachen : Shaker Verlag, 2002. – ISBN 3–8265–9924–1
- [Pahl und Beucke 2000] PAHL, P. J. ; BEUCKE, K.: Neuere Konzepte des CAD im Bauwesen: Stand und Entwicklungen. In: *Internationales Kolloquium über Anwendungen der Informatik und Mathematik in Architektur und Bauwesen (IKM)*, Digital Proceedings, 2000
- [Pahl und Damrath 2000] PAHL, P. J. ; DAMRATH, R.: *Mathematische Grundlagen der Ingenieurinformatik*. Berlin, Heidelberg, New York : Springer, 2000. – ISBN 3–540–60501–0